



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/662,588	09/15/2000	Fred Irwin	CITI0184	1954

27510 7590 12/19/2005

KILPATRICK STOCKTON LLP  
607 14TH STREET, N.W.  
WASHINGTON, DC 20005

EXAMINER
----------

BORLINGHAUS, JASON M

ART UNIT	PAPER NUMBER
----------	--------------

3628

DATE MAILED: 12/19/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

**Office Action Summary**

Application No.

09/662,588

Applicant(s)

IRWIN ET AL.

Examiner

Jason M. Borlinghaus

Art Unit

3628

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --  
Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

- 1) ☒ Responsive to communication(s) filed on 26 September 2005.  
2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.  
3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

- 4) ☒ Claim(s) 8-27 is/are pending in the application.  
4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.  
5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.  
6) ☒ Claim(s) 8-27 is/are rejected.  
7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.  
8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

**Application Papers**

- 9) ☐ The specification is objected to by the Examiner.  
10) ☐ The drawing(s) filed on \_\_\_\_\_ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).  
11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).  
a) ☐ All b) ☐ Some \* c) ☐ None of:  
1. ☐ Certified copies of the priority documents have been received.  
2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.  
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

- 1) ☒ Notice of References Cited (PTO-892)  
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)  
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)  
Paper No(s)/Mail Date \_\_\_\_\_.  
4) ☐ Interview Summary (PTO-413)  
Paper No(s)/Mail Date. \_\_\_\_\_.  
5) ☐ Notice of Informal Patent Application (PTO-152)  
6) ☐ Other: \_\_\_\_\_.

## DETAILED ACTION

### ***Claim Rejections - 35 USC § 103***

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

The factual inquiries set forth in *Graham v. John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966), that are applied for establishing a background for determining obviousness under 35 U.S.C. 103(a) are summarized as follows:

1. Determining the scope and contents of the prior art.
2. Ascertaining the differences between the prior art and the claims at issue.
3. Resolving the level of ordinary skill in the pertinent art.
4. Considering objective evidence present in the application indicating obviousness or nonobviousness.

**Claims 8 – 27** are rejected under 35 U.S.C. 103(a) as being unpatentable over Kalmus (US Patent 4,674,044) in view of Schlidt (Schildt, Herbert. *Turbo C/C++: The Complete Reference*. Osborne McGraw-Hill. Berkeley, CA. 1990. pp. 13, 561 and 727 - 730) and (Coughlin, George Gordon. *Your Handbook Of Everyday Law*. 5<sup>th</sup> Edition. Harper Collins Publishing. New York, NY. 1993. pp. 50 – 51).

**Regarding Claim 8**, Kalmus discloses a system comprising:

- a customer terminal (brokerage firm's account executives and/or computer equipped customers). (see col. 4, lines 60 – 69);

Art Unit: 3628

- a trader terminal (see 15, figure 1) operatively coupled to the customer terminal (see 27 and 29, figure 1) through a computer network (11, 25 and 26, figure 1);
- a processor (CPU – see 10, figure 1);
- wherein the processor (CPU) is involved in a trade request (order generated) from a customer at a customer terminal (brokerage firm's account executives and/or computer equipped customers - see col. 4, lines 60 – 69) further comprising:
  - a first component comprising functions for sending messages and receiving messages to the system (CPU – see 10, figure 1) on behalf of the customer (Brokerage House – see 27, figure 1). ("Input/output network 25 provides data communication with the various branch offices 27 of the brokerage house. Line 25 permits communication with either the branch order entry clerk or directly to the account executives at each branch." – see col. 4, lines 41 – 45 - It is inherent that a component comprises the functions for sending messages and receiving messages to the system on behalf of a customer.);
  - a second component comprising functions for controlling access to the system by the customer. ("The order data fields include ... customer identification (CUSTID)..." – see col. 5, lines 52 – 54 – It is inherent that a component comprises functions for controlling access to the system by the

customer since the functioning of the system requires a valid customer identification); and

- a third component comprising functions for sending messages to and receiving messages from the first component and a trader (Trader – see 15, figure 1) at the trader terminal. (see 11, figure 1 - It is inherent that a component comprises functions for sending messages to and receiving messages from the first component and a trader.)

Kalmus does not teach a system comprising:

- a processor is configured to dynamically create sets of class components to handle one or more transactions with each set of class components;  
and
- wherein the processor comprises a timer wherein the trade request from the customer is automatically revoked at a predetermined duration of time if the trader does not accept the trade request.

Object-oriented programming and class-based programming are old and well known in the art for designing, modeling, building and developing software systems, as evidenced by Schildt who states that object-oriented programming and class-based programming, an object-based programming language, has been in use since 1980 (“In 1980, while working at Bell Laboratories at Murray Hill, New Jersey, Bjarne Stroustrup addressed this problem by adding several extensions to the C language. Initially, called ‘C with Classes,’ the name was changed to C++ in 1983.” – see page 727). It would have been obvious to a person of ordinary skill in the art to have modified Kalmus to

Art Unit: 3628

allow for his trading system to be encoded or programmed in any computer language that the inventor desired, including object-oriented programming language or class-based programming language which were already in use at the time of invention.

Furthermore, an object-based programming language and a class-based programming language, which were both known at the time of the invention, would have been obvious choices by which to reinterpret Kalmus due to the numerous benefits of such programming languages – easy programming maintenance, easy to understand and streamlined structure, as evidenced by Schildt who states, “Object-oriented programming allows you to easily decompose a problem into subgroups of related parts. Then you translate these subgroups into self-contained units called objects” (see page 729) and “Also, because C++ shares C’s efficiency, high-performance systems software can be constructed using C++.” (see page 728).

Setting a time limit for acceptance of an offer and the revocation of said offer if not accepted within the established time limit is old and well known in the art of contract formation, as evidenced by Coughlin who states “For how long is an offer good, and how is an offer terminated? Generally, an offer is deemed to be terminated ... (2) by the lapse of the time specified or the lapse of a reasonable time when the offer is silent concerning duration.” (see Termination of Offer, p. 50). It would have been obvious to one of ordinary skill to have modified Kalmus and Schildt by incorporating into the processor a timer which would revoke the trade request (offer) when such trade request (offer) was not accepted within the predetermined duration of time, as is old and well known, to account for time limitations on trade requests (offers) sent to traders.

**Regarding Claims 9 – 14**, Kalmus discloses a system wherein:

- the third component operates in a synchronous (real-time) format. (“The market making system of the above-described invention has thus been shown to automatically accommodate a random, real time order flow for security purchases or sales.” – see col. 10, lines 23 – 26);
- the third component operates in a asynchronous (time delay executability) format. (“Orders not qualified for execution are stored and re-examined from time to time for possible later executability.” – see col. 10, lines 34 – 36); and
- components are configured to handle multiple customers at one time. (“While only one branch is shown in Figure 1, it is to be understood that a multiplicity of branches 27 are in data communication with processor 10.” – see col. 4, lines 45 – 48 – It is inherent that the set of class components are configured to handle multiple customers at one time).
- components are configured to handle multiple transactions at one time. (“While only one branch is shown in Figure 1, it is to be understood that a multiplicity of branches 27 are in data communication with processor 10.” – see col. 4, lines 45 – 48 – It is inherent that the components are configured to handle multiple transactions at one time with multiple customers in data communication with the processor.).

Kalmus does not teach a system wherein:

Art Unit: 3628

- the set of class components are configured to handle multiple customers at one time;
- the set of class components are configured to handle a single customer at one time;
- the set of class components are configured to handle a single transaction at one time; and
- the set of class components are configured to handle multiple transactions at one time.

However, functionality can be deleted from Kalmus to reduce the its ability to handle multiple customers and multiple transactions at one time:

- to handle a single customer at one time; and
- to handle a single transaction at one time.

It would have been obvious to one of ordinary skill in the art at the time the invention was made to have modified Kalmus by incorporating the ability to handle both multiple and single customers, and both multiple and single transactions, to allow for versatility in the operation and functioning of the system.

Additionally, as discussed above in Claim 8 and is evidenced by Schildt, object-oriented programming and class-based programming are old and well known in the art for designing, modeling, building and developing software systems.

Accordingly, at the time of the invention, it would have been obvious to a person of ordinary skill in the art to have modified Kalmus to allow for his trading system to be encoded or programmed in any computer language that the inventor desired, including



Art Unit: 3628

object-oriented programming language or class-based programming language which were already in use at the time of invention was made, to reap the benefits of that computer programming language, as discussed above.

**Regarding Claim 15**, Kalmus does not teach a system wherein:

- the processor creates sets of class components based on the number of transactions.

However, as discussed above in Claim 8 and is evidenced by Schildt, object-oriented programming and class-based programming are old and well known in the art for designing, modeling, building and developing software systems.

Therefore, at the time of the invention, it would have been obvious to a person of ordinary skill in the art to have modified Kalmus to allow for his trading system to be encoded or programmed in any computer language that the inventor desired, including object-oriented programming language or class-based programming language which were already in use at the time of invention was made, to reap the benefits of that computer programming language, as discussed above.

Additionally, it is old and well known in the art that software creates and assigns memory space (an object, a memory location, a file) to store customer information and customer transaction data during customer login and transaction execution. As evidenced by Schildt who states memory is allocated during transaction execution "In this method, storage for information is allocated from the free memory area as it is needed and returned to free memory when it has served its purpose...Because memory can be allocated for one purpose and freed when that use has ended, it is possible for

Art Unit: 3628

another part of the program to use the same memory for something else at a different time.” (see page 561).

Therefore, at the time of the invention, it would have been obvious to a person of ordinary skill in the art to have modified Kalmus to allow for the processor to create class components based on the number of transactions since, as evidenced by Schildt, memory space would need to be allocated based on the number of transactions to be executed by the system, as is standard in software programming.

**Regarding Claim 16**, further method claim would have been obvious from system claim rejected above, Claim 8, and is therefore rejected using the same art and rationale.

**Regarding Claim 17**, Kalmus does not teach a method wherein:

- each component is created in response to a customer accessing the system.

It is old and well known in the art that software creates and assigns memory space (an object, a memory location, a file) to store customer information and customer transaction data during customer login and transaction execution. As evidenced by Schildt who states memory is allocated during transaction execution “In this method, storage for information is allocated from the free memory area as it is needed and returned to free memory when it has served its purpose...Because memory can be allocated for one purpose and freed when that use has ended, it is possible for another part of the program to use the same memory for something else at a different time.” (see page 561).

Art Unit: 3628

Therefore, at the time of the invention, it would have been obvious to a person of ordinary skill in the art to have modified Kalmus to create components in response to a customer accessing the system since, as evidenced by Schildt, memory space would need to be allocated based on system access by customers, as is standard in software programming.

**Regarding Claim 18**, Kalmus discloses a computer program comprising:

- at least one computer-readable medium. (It is inherent that the system would require software programming contained on a computer-readable medium to function); and
- a programming module stored on the at least one medium, and operable, upon access of a customer to trading services of the computer program product for handling one or more transactions involving a trade request from the customer to a trader (“The order data fields include ... customer identification (CUSTID)...” – see col. 5, lines 52 – 54 – It is inherent that a valid customer identification is required to allow access to trading services) to;
- where created programming include at least one of:
  - an access control programming. (“The order data fields include ... customer identification (CUSTID)...” – see col. 5, lines 52 – 54 – It is inherent that an access control programming must exist in order to validate customer identification);

Art Unit: 3628

- a trading system communications programming. (see 11, 12, 22, 23, 25 and 26, figure 1 - It is inherent that a trading system communications programming must exist to manage communications for the trading system); and
- a translator programming (“...data input apparatus such as a keyboard...” – see col. 4, lines 9 – 10 – It is inherent that a translator programming must exist to convert customer input into executable transactions.)

Kalmus does not teach a computer program comprising:

- a class creation module stored on the at least one medium, and operable, upon access of a customer to trading services of the computer program product, to create at least one set of classes, each set comprising at least one class;
- where created classes include at least one of:
  - an access control class;
  - a trading system communications class; and
  - a translator class; and
- a timer module stored on the at least one medium; and
- operable to automatically revoke at a predetermined time the trade request from the customer if the trader does not accept the trade request.

However, as discussed above in Claim 8 and is evidenced by Schildt, object-oriented programming and class-based programming are old and well known in the art for designing, modeling, building and developing software systems.

Therefore, at the time of the invention, it would have been obvious to a person of ordinary skill in the art to have modified Kalmus to allow for his trading system to be encoded or programmed in any computer language that the inventor desired, including object-oriented programming language or class-based programming language which were already in use at the time of invention was made, to reap the benefits of that computer programming language, as discussed above.

Furthermore, as discussed above in Claim 8 and is evidenced by Coughlin, setting a time limit for acceptance of an offer and the revocation of said offer if not accepted within the established time limit is old and well known in the art of contract formation.

Therefore, it would have been obvious to one of ordinary skill to have modified Kalmus and Schildt by incorporating into the processor a timer which would revoke the trade request (offer) when such trade request (offer) was not accepted within the predetermined duration of time, as is old and well known, to account for time limitations on trade requests (offers) sent to traders.

**Regarding Claims 19 – 20**, further computer program claims would have been obvious from system claim rejected above, Claims 13- 14, and are therefore rejected using the same art and rationale.

**Regarding Claim 21**, Kalmus does not teach a computer program where:

- each class being an object linking and embedded class type.

However, as discussed above in Claim 8 and is evidenced by Schildt, object-oriented programming and class-based programming are old and well known in the art for designing, modeling, building and developing software systems. Furthermore, object-linking and embedded class types are standard in object-oriented programming and class-based programming languages. As evidenced by Schildt that discusses object linking by stating "Later the linker combines the code you wrote with the object code already found in the standard library. This process is called linking." (see page 13).

Therefore, at the time of the invention, it would have been obvious to a person of ordinary skill in the art to have modified Kalmus to allow for his trading system to be encoded or programmed in any computer language that the inventor desired, including object-oriented programming language or class-based programming language which were already in use at the time of invention was made, and utilizing all techniques commonly employed by that programming language, to reap the benefits of that computer programming language, as discussed above.

Furthermore, as discussed above in Claim 8 and is evidenced by Coughlin, object-oriented programming and class-based programming are old and well known in the art for designing, modeling, building and developing software systems. Furthermore, object-linking and embedded class types are standard in object-oriented programming and class-based programming languages. As evidenced by Schildt that discusses object linking by stating "Later the linker combines the code you wrote with the object code already found in the standard library. This process is called linking." (see page 13).

Art Unit: 3628

**Regarding Claim 22**, further computer program claims would have been obvious from a subsection of the program claim rejected above, Claim 18, and is therefore rejected using the same art and rationale.

**Regarding Claim 23**, further method claim would have been obvious from a subsection of the program claim rejected above, Claim 18, and is therefore rejected using the same art and rationale.

**Regarding Claims 24 - 25**, further method claims would have been obvious from system claim rejected above, Claims 13- 14, and are therefore rejected using the same art and rationale.

**Regarding Claim 26**, further method claim would have been obvious from computer claim rejected above, Claim 21, and is therefore rejected using the same art and rationale.

**Regarding Claim 27**, further method claim would have been obvious from a subsection of the program claim rejected above, Claim 18, and is therefore rejected using the same art and rationale.

### ***Response to Arguments***

Applicant's arguments with respect to pending claims have been considered but are moot in view of the new ground(s) of rejection.

### ***Conclusion***

Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Jason M. Borlinghaus whose telephone number is (571) 272-6924. The examiner can normally be reached on 8:30am-5:00pm M-F.


If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Hyung Sough can be reached on (571) 272-6799. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.



Art Unit: 3628

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

\*\*\*

  
FRANTZY POINVIL  
PRIMARY EXAMINER  
Au 3628